

Lección 27: El proceso de diseño de juego

Lección en línea [Ver en Code Studio](#)

Propósito

Esta Lección presenta a los estudiantes el proceso que usarán para diseñar juegos. Este proceso se centra en una guía de proyecto que les pide a los estudiantes que definan sus sprites, variables y funciones antes de comenzar a programar su juego. En esta Lección, los estudiantes comienzan jugando un juego en Game Lab donde el código está oculto. Discutan qué creen que deberían hacer los sprites, las variables y las funciones para hacer el juego. Luego reciben una guía de proyecto completa que muestra una forma de implementar el juego. A continuación, los estudiantes pasan por este proceso a través de una serie de niveles. Como parte de esta Lección, los estudiantes también aprenden brevemente a utilizar animaciones de cuadros múltiples en Game Lab. Al final de la Lección, los estudiantes tienen la oportunidad de hacer mejoras en el juego para hacerlo suyo.

Esta Lección presenta animaciones de cuadros múltiples y es la primera de una secuencia centrada en el proceso de creación de software.

Si bien las lecciones anteriores se centraron en el uso de abstracciones para gestionar la complejidad del código, esta Lección se centra en la gestión de la complejidad del proceso de desarrollo de software. La Lección entrega en gran medida el proceso de desarrollo de software al proporcionar a los estudiantes una guía de proyecto completa, proporcionando código de inicio, y encaminando a los estudiantes a través de su implementación. En las lecciones posteriores, los estudiantes deberán completar una mayor parte de esta guía de forma independiente, y para el proyecto final, seguirán este proceso.

Secuencia para el aprendizaje

- Conocimiento inicial (15 min)
- Ampliación del conocimiento (60 min)
- Transferencia del conocimiento (20 min)

Objetivos

Los estudiantes serán capaces de:

- Identificar las construcciones de programación central necesarias para construir diferentes componentes de un juego.
- Crear y usar animaciones de multitarea en un programa.
- Implementar diferentes características de un programa siguiendo una guía de proyecto estructurado.

Preparación

- [Contenido de la Lección - code.org](#)

Recursos

¡Atención!
Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Defender Game - Guía de proyectos](#)

Estrategia de aprendizaje

Conocimiento inicial (15 min)

Jugar Cake Defender

Esta Lección comienza inmediatamente en Code Studio [Contenido de la Lección - code.org](#). En el primer nivel, encontrarán un juego, pero no podrán ver el código. Deben jugar el juego y seguir las instrucciones que les piden que enumeren las variables, los sprites y las funciones que creen que son necesarios para crear este juego.

¡Defiende tu Torta!

Es un ejemplo de un juego que construirás al final de esta Lección.

Detener: revisar la guía de proyectos

Los estudiantes deberán crear individualmente una lista de variables, sprites y funciones que crean que se utilizan en el juego. Pida a los estudiantes que compartan sus listas con un compañero antes de hablar como clase.

Distribuir:

Entregue a cada alumno o pareja de estudiantes una copia del Juego Defender - Guía de proyectos

Preguntar:

Compare los componentes del juego que creían que se incluirían con los de esta guía de proyecto. ¿Notan alguna diferencia?

Discuta:

Como clase, compara la lista que tenías en el pizarrón con la lista de variables, sprites y funciones en la guía del proyecto. Tenga en cuenta las similitudes. Donde hay diferencias que intenten entender por qué. Evite “correcto” o “incorrecto”.

Por lo general, hay muchas maneras de estructurar un programa para que funcione de la manera que desea. Lo importante al escribir programas complejos o grandes, es que comienzan con un plan. Hoy vamos a ver cómo podemos implementar este plan para construir nuestro propio juego. Al final de la Lección, no sólo habrán construido un juego, sino que sabrán cómo cambiarlo y hacerlo propio. ¡Vamos!

Mantener el enfoque: los estudiantes pueden distraerse fácilmente con la diversión de jugar el juego. Déjelos jugar por un tiempo, pero eventualmente animelos a seguir las instrucciones en pantalla y haga una lista de las variables, los sprites y las funciones que sería necesarias para jugar el juego.

Ejecución de la conversación: puede escribir “variables”, “sprites” y “funciones” en la pizarra y registrar sus ideas debajo de cada una. Pídale a los estudiantes que justifiquen sus decisiones, pero no sientan la necesidad de conformarse con una sola respuesta correcta

Guía del proyecto: la guía del proyecto se completa de manera exhaustiva para los estudiantes, de modo que pueda experimentar su uso como referencia cuando programe. Esto debería darles más contexto al completar su propia guía de proyectos en las próximas dos lecciones.

Puede dar a cada uno su propia copia como referencia, pero también puede optar por imprimir una copia por par, compartir copias digitales o simplemente mostrar la guía en el proyector. Siempre que esté disponible para referencia, cualquier enfoque funcionará bien.

Ampliación del conocimiento (60 min)

Animaciones de multitrama

Transición:

Los estudiantes deben regresar a Code Studio. Antes de implementar realmente el plan, los estudiantes deberán revisar rápidamente una nueva habilidad, cómo usar animaciones de multitrama. Los estudiantes aprenderán rápidamente a crear, modificar y cambiar el nombre de las animaciones, así como a elegir las de la biblioteca.

Detener

Antes de seguir adelante, deberán consultar la Guía del proyecto.

Implementar la guía del proyecto

Los estudiantes reciben una gran cantidad de código de inicio en este proyecto. Los sprites, las variables y las funciones ya se les han asignado. El trabajo de este proyecto es escribir el código para las funciones individuales. Estos niveles guían a los estudiantes a través de cómo implementar esas funciones. A medida que los estudiantes avanzan por los niveles, señalan cómo se está utilizando la guía del proyecto.

Las habilidades más desafiantes que usan los estudiantes en estos niveles es reconocer la necesidad de crear nuevas funciones para reemplazar el código repetido. Los estudiantes necesitan desarrollar esta habilidad por su cuenta, pero estos niveles demuestran una instancia donde esto podría suceder.

Realizar los niveles de Code Studio correspondiente a la Lección.

Transferencia del conocimiento (20 min)

Hagalo propio

Este último nivel alienta a los estudiantes a hacer suyo el juego. Si los estudiantes han llegado a este punto, tienen todas las habilidades que necesitan para progresar a través del plan de estudios, por lo que no hay presión para completar ninguna de las modificaciones sugeridas en este nivel. Sin embargo, si tienen tiempo, obtener una práctica de planificación e implementar nuevas características será una habilidad útil. Incluso modificar las animaciones del juego es una forma sencilla de que los estudiantes puedan apropiarse del juego.

Compartir:

Una vez que los estudiantes han completado el proyecto, pueden compartir su trabajo con sus compañeros de clase. Alienta a los estudiantes a exhibir el código adicional que escribieron y explica cómo ha cambiado la forma en que funciona el juego.

Sugerencias para evaluar

Se sugiere los siguientes indicadores para evaluar formativamente los aprendizajes:

- Elaboran una guía de programa
- Explican la finalidad de las diferentes estructuras utilizadas
- Colaboran de forma grupal en el desarrollo del diseño del juego